WE CLAIM:

1. A computer processing method comprising the steps of:

creating at least one logic web of at least one software entity ("molecule") which is configured with software micro-components including a signal handler, at least one input handler, at least one output handler, an interface handler, at least one method handler and an associated method, said signal handler having means for sending and receiving communication signals externally of the respective molecule and being operatively connected to the other micro-components, said at least one input handler having means for queuing input data, said interface handler having means for determining when predefined input conditions for the presence of required data is fulfilled and invoking said method handler, said method handler having means for invoking said associated method for processing the input data, and said at least one output handler having means for outputting a result of the processing of input data by said method, and

deploying the at least one logic web with a computing resource having means for loading the at least one molecule into random access memory and running the software entity as a computer processing task.

2. A computer processing method according to Claim 1, wherein said creating step includes creating a plurality of logic webs each having its web of software entities configured to perform a data processing task with a computing resource autonomously, and said deploying step includes deploying the plurality of logic webs with respective ones of a plurality of computing resources, respectively.

3. A computer processing method according to Claim 2, wherein said plurality

of computing resources are distributed on a network in a distributed computing environment.

4. A computer processing method according to Claim 2, wherein said plurality of computing resources include an array of central processing units (CPUs) in parallel in a parallel processing environment.

5. A computer processing method according to Claim 1, wherein the at least one logic web is comprised of a plurality of molecules, and said logic web exists as initialization files for generating an initial host of molecules which invoke and generate other molecules in successive layers of incremental processing steps.

6. A software programming method comprising the steps of:

creating at least one software entity ("molecule") which is configured with software micro-components including a signal handler, at least one input handler, at least one output handler, an interface handler, at least one method handler and an associated method, said signal handler having means for sending and receiving communication signals externally of the respective molecule and being operatively connected to the other micro-components, said at least one input handler having means for queuing input data, said interface handler having means for determining when predefined input conditions for the presence of required data is fulfilled and invoking said method handler, said method handler having means for invoking said associated method for processing the input data, and said at least one output handler having means for outputting a result of the processing of input data by said method, and

assigning said at least one software entity to a selected computing resource and to use a source of input data.

7. A software programming method according to Claim 6, wherein said creating step includes creating an initialization file for the at least one molecule by specifying its software micro-components in a selected programming language.

8. A software programming method according to Claim 6, further comprising the step of maintaining a library of software micro-components in a plurality of handler types identified by name, and wherein said creating step includes invoking the names of selected handler types for the software micro-components of the molecule to be retrieved from said library.

9. A software programming method according to Claim 6, wherein said interface handler includes means for providing the molecule with the characteristic of autonomously waiting, looking, and proceeding with said associated method for processing the input data by waiting until said input handler indicates that the predefined input conditions are present before invoking said method handler for the associated method.

10. A software programming method according to Claim 6, wherein said interface handler includes a plurality of means for determining when respective predefined input conditions for the presence of respectively required data is fulfilled and for invoking respective ones of a plurality of method handlers and associated methods.

11. A software programming method according to Claim 6, wherein said input handler is selected from one of a plurality of input handler types corresponding respectively

to a plurality of different data source types.

12. A software programming method according to Claim 6, wherein a plurality of molecules are created in said creating step and configured in a logic web to perform a data processing task.

13. A software programming method according to Claim 12, wherein said creating step includes creating molecules having means for creating next molecules in successive layers of incremental processing steps.

14. A software programming method according to Claim 6, wherein said creating step includes creating molecules having built-in means for performing a clean-up of its functions when the molecule is to be terminated.

15. A software programming method according to Claim 6, wherein said creating step includes creating molecules having means for recording information on the state of its micro-component handlers and signaling such state information externally through said signal handler.

16. A software programming method according to Claim 6, wherein said signal handler can receive signals for and has means for dynamically reconfiguring the micro-component handlers of the molecule while it is in existence to perform a processing task.

17. A software entity ("molecule") comprising, as software micro-components: a signal handler,

at least one input handler,

at least one output handler,

an interface handler,

at least one method handler and

an associated method,

wherein said signal handler includes means for sending and receiving communication signals externally of the respective molecule and for operatively connecting to the other micro-components, said at least one input handler includes means for queuing input data, said interface handler includes means for determining when predefined input conditions for the presence of required data is fulfilled and invoking said method handler, said method handler includes means for invoking said associated method for processing the input data, and said at least one output handler includes means for outputting a result of the processing of input data by said method, and

wherein the software entity is assigned to a selected computing resource and to use a source of input data.

18. A software entity according to Claim 17, wherein said interface handler includes means for providing the molecule with the characteristic of autonomously waiting, looking, and proceeding with said associated method for processing the input data by waiting until said input handler indicates that the predefined input conditions are present before invoking said method handler for the associated method.

19. A software entity according to Claim 17, wherein said interface handler includes a plurality of means for determining when respective predefined input conditions for the presence of respectively required data is fulfilled and for invoking respective ones of

a plurality of method handlers and associated methods.

20. A software entity according to Claim 17, wherein said input handler is selected from one of a plurality of input handler types corresponding respectively to a plurality of different data source types.